

Article: Sequential Control

Finn Haugen
TechTeach

22. August 2009

Preface

This article gives an introduction to sequential control using Sequential function charts (SFCs) and State diagrams, and it can be downloaded for free from <http://techteach.no>. It can be used as supplementary material to the book *Basic Dynamics and Control*, which can be purchased from <http://techteach.no>.

1 Introduction

Sequential control is crucial in many control applications, as in chemical batch processes. A sequential control procedure can be represented graphically by e.g. one of the following two methods:

- A State diagram, also denoted a State machine
- A Sequential function chart (SFC)

State diagrams and SFC are quite similar, as they have the same basic elements:

- States with actions to be executed when the state is active
- Transitions from one active state to another taking place if the transition condition is logically true.

Controllers can be implemented using graphical programming of state diagrams or SFCs in proper programming tools. For example, state diagrams are supported by the State Diagram Toolkit in LabVIEW, and HiGraph in the Step 7 programming tools of Simatic PLCs, while SFC is

supported by the GX IEC Developer programming tool of Mitsubishi FX2N PLCs.

2 State diagrams

This document gives an introduction to state diagrams. State diagrams can be used to define sequential control. State diagrams are quite similar to Sequential function charts (SFCs). SFC is standardized in the IEC 61131-3 standard (Introduction to IEC 61131-3), and is available as a programming tool in most PLC systems, e.g. Mitsubishi PLCs, Simatic PLCs, etc. State diagrams are supported in e.g. State Diagram Toolkit of LabVIEW, the Stateflow Toolbox of Matlab/Simulink, and the Graph7 programming tool of Simatic PLCs.

The symbols and terms used in this section is not according to any specific standard, but they are similar to symbols and terms used in practical computer tools for creating state diagrams (some are mentioned above).

Figure 1 shows the elements of state diagrams. (The figure shows one example of a state diagram. Other diagrams may be simpler or more complicated.)

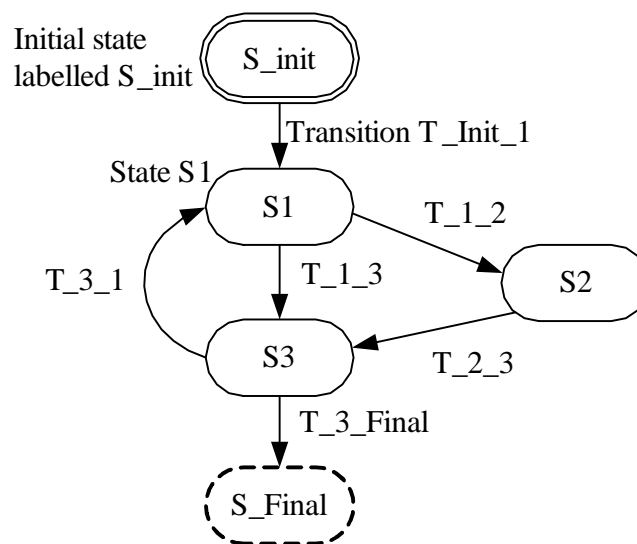


Figure 1: Elements of state diagrams: States and transitions

A state diagram represents a *state machine*. A state machine has a number

of *states*. Each state is either active or passive. Only one state is active at a time. The state machine always starts in a particular state defined as the initial state, and it ends (stops) after the final state. Special symbols or colors may be used for the initial state and the final state, cf. Figure 1.

Associated with a state are one or more *actions*. An action may be a control action executed by a controller. Actions may be listed as indicated below:

State 1 actions:

Action 1: Valve V1 is open

Action 2: Motor M1 runs

State 2 actions:

Action 1: Valve V1 is closed

Action 2: Heater H1 is on

The actions for a given state can be listed in a box attached to the state symbol, see Figure 2. However, the diagram may then become overloaded by symbols. Therefore, in more comprehensive state diagrams, it may be better to just list the actions in a separate document.

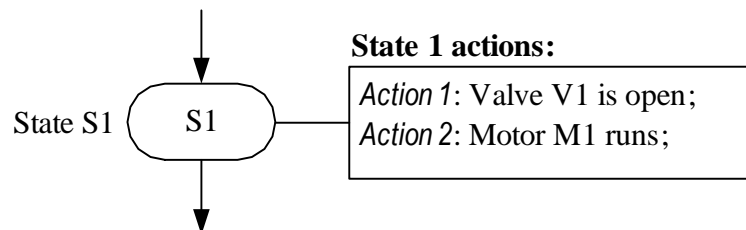


Figure 2: Action box listing actions of a state

Transitions brings the state machine from one state to another. A transition can go only from an active state which is presently active. A transition takes place only if its transition condition is TRUE. Transition conditions are in the form of logical expressions having value either TRUE or FALSE and may be listed as indicated below:

- Transition T_Init_1: Button_start = ON.
- Transition T_1_2: L1 > Level_High;

where Transition T_1_2 means the transition from state S1 to state S2. A transition condition may be written directly in the state diagram. For example, the expression

$$T_Init_1: Button_start = ON$$

may be written close to the transition arrow.

A transition may take place unconditionally, that is, it takes place immediately after the actions of the presently active state have been accomplished. Such unconditional transitions are named default transitions. The transition condition of an unconditional (or default) transition has permanent value TRUE, and it may be expressed as indicated below:

$$T_3_4: Default$$

or

$$T_3_4: TRUE$$

3 Sequential function charts (SFCs)

Sequential function charts (SFCs) are similar to State diagrams. One small difference is that a transition condition is indicated by a small line in the transition path from one state to another. And, typically, the states are denoted steps, which are represented by rectangles, see Figure 3.

Eksempel 1 *Control system of a batch process*

Figure 4 shows a simple batch process which is to be controlled by sequential control. (A similar simulator of this system is available at <http://techteach.no/simview>.) The tank is filled with water. The water is then heated and stirred¹, and finally the heated water is discharged from the tank.

The *control signals* are

¹The motor is assumed to ensure homogenous thermal conditions in the water in the tank.

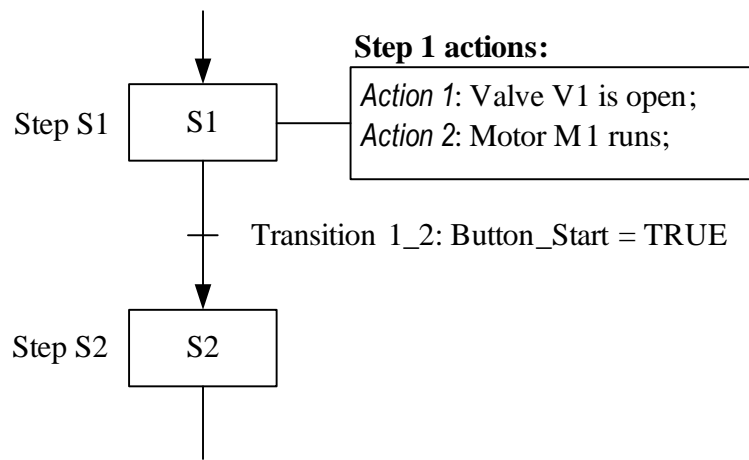


Figure 3: Elements of a Sequential Function Chart (SFC)

- u_valve_in (boolean, i.e. having value TRUE or FALSE, or ON or OFF)
- u_valve_out (boolean)
- u_motor (boolean)
- u_heat (continuous having any value between 0% and 100%)

The *measurements* are

- Temperature T of the water in the tank
- Level h of the water in the tank

Figure 5 shows an overall block diagram of the control system defining setpoints (specified values), control signals, and measurements. Figure 6 shows a SFC (Sequential Function Chart) defining the control task.

Figure 7 shows the state diagram defining the sequential control of the batch process using the State diagram toolkit of LabVIEW. (This state diagram picture is generated by the toolkit.)

[End of Example 1]

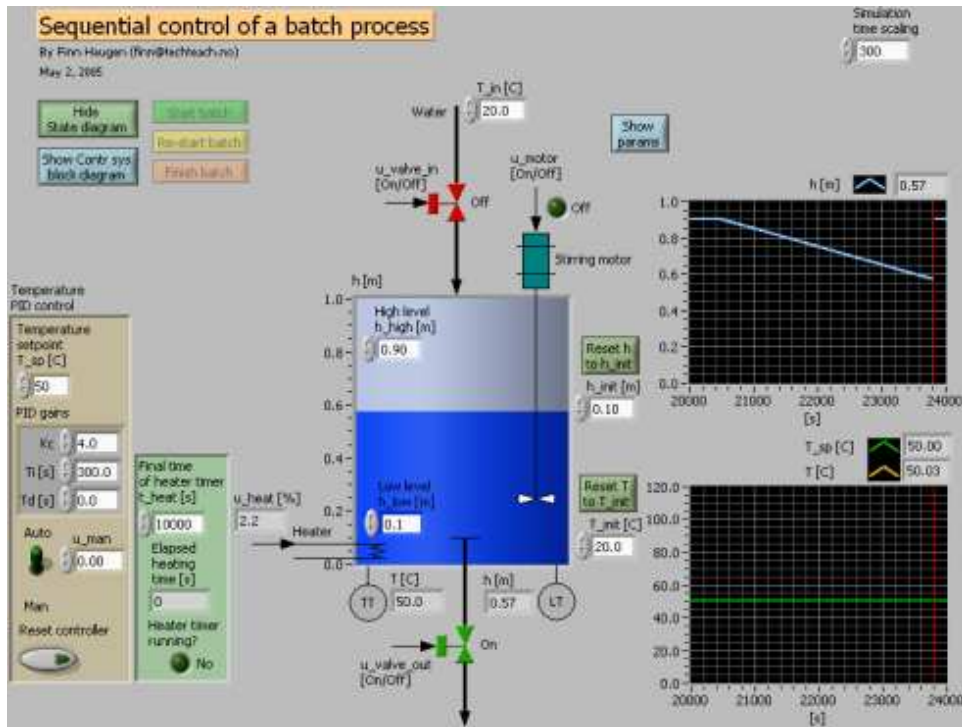


Figure 4: A batch process to be controlled by sequential control

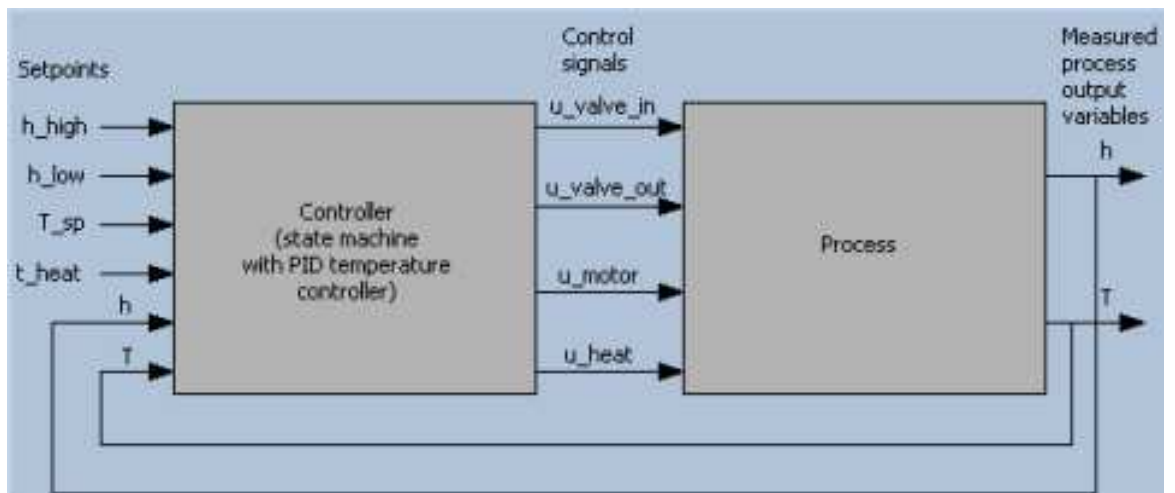


Figure 5:

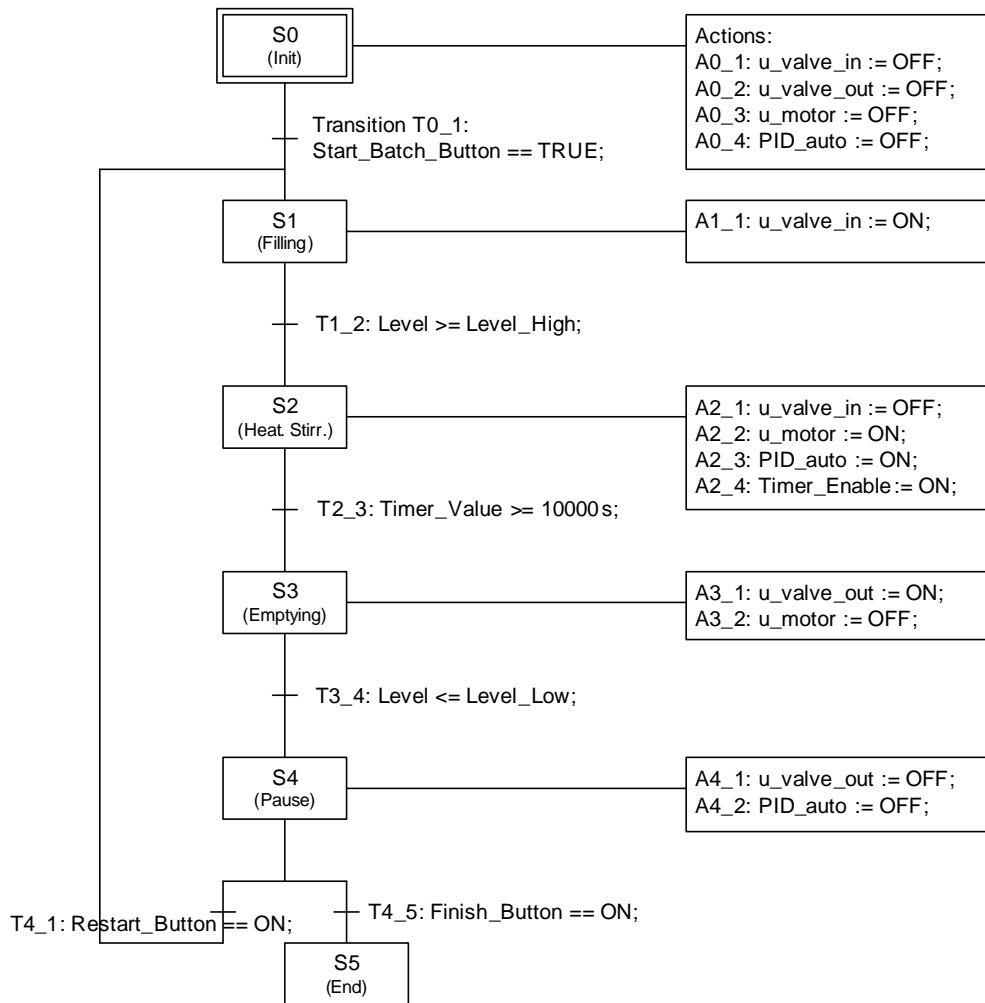


Figure 6: Sequential Function Chart (SFC) defining the control

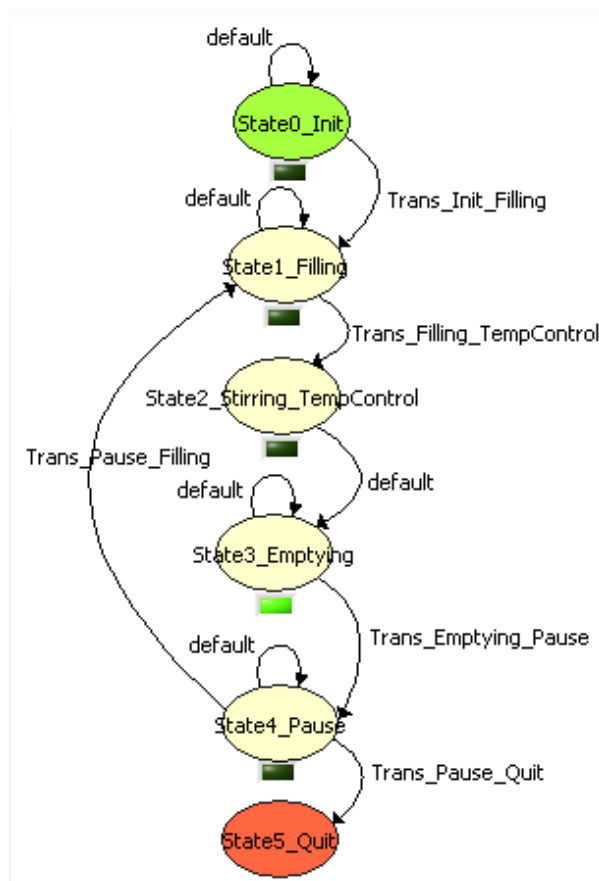


Figure 7: The state diagram defining the sequential control of the batch process using the State diagram toolkit of LabVIEW